CLAIMS

We claim:

1. A decoder to correct errors in data, comprising:

a plurality of cells, each cell configured to generate a partial syndrome based on data blocks and one or more redundancy blocks, and to generate a partial error value based on the partial syndrome and a portion of an inverse of an error location matrix that identifies locations of any data blocks having errors; and

a summation logic connected to the plurality of cells, the summation logic configured to generate an error value based on the partial error values generated by the plurality of cells, wherein the error value corrects errors in one of the data blocks.

2. The decoder of claim 1, wherein the number of cells in the plurality of cells is equal to the number of redundancy blocks.

3. The decoder of claim 1, wherein each cell includes:

an input logic;

a first queue connected to the input logic; and

a multiplier connected to the input logic and the first queue.

4. The decoder of claim 1, wherein each cell operates in a read mode to generate the partial syndrome.

5. The decoder of claim 4, wherein when a cell operates in a read mode, the input logic of the cell is set to receive inputs from the multiplier and a data buffer that holds the data blocks and one or more redundancy blocks.

6. The decoder of claim 5, wherein the input logic generates a sum of the inputs from the multiplier and the data buffer, and wherein the sum is stored in the first queue.

7. The decoder of claim 6, wherein the input logic includes an exclusive-or (XOR) gate with inputs connected to the data buffer and the multiplier and an output connected to the first queue to generate the sum.

8. The decoder of claim 6, wherein when the cell operates in the read mode, the multiplier is set to receive a root of a generator polynomial and the sum from the first queue.

9. The decoder of claim 8, wherein the multiplier is a Galois Field multiplier that performs a Galois Field multiplication between the root of a generator polynomial and the sum.

10. The decoder of claim 9, wherein the multiplier comprises a plurality of Galois Field multipliers to perform multiple, parallel Galois Field multiplications between the root of a generator polynomial and the sum.

11. The decoder of claim 8, wherein each cell in the plurality of cells is set to receive a different root of a generator polynomial to generate different partial syndromes.

12. The decoder of claim 8, wherein the root of a generator polynomial is a hardwired constant or a register set by firmware.

13. The decoder of claim 8, further comprising:

a second queue that holds the root of a generator polynomial; and

a multiplexer having an input connected to the second queue and an output connected to the multiplier, wherein the multiplexer is set to connected the second queue to the multiplier when the cell operates in the read mode.

14. The decoder of claim 4, wherein each cell operates in a write mode to generate the partial error value.

15. The decoder of claim 14, wherein when a cell operates in a write mode, the input logic of the cell is set to receive inputs from the queue that holds the partial syndrome generated by the cell during the read mode.

16. The decoder of claim 15, wherein the multiplier is a Galois Field multiplier that performs a Galois Field multiplication between the partial syndrome and the element of the portion of the inverse of the error location matrix.

17. The decoder of claim 16, wherein the multiplier comprises a plurality of Galois Field multipliers to perform multiple, parallel Galois Field multiplications between the partial syndrome and the element of the portion of the inverse of the error location matrix.

18. The decoder of claim 14, wherein each cell in the plurality of cells is set to receive a different portion of the error location matrix to generate different partial error values.

19. The decoder of claim 14, further comprising:

a third queue that holds the portion of the error location matrix; and

a multiplexer having an input connected to the third queue and an output connected to the multiplier, wherein the multiplexer is set to connected the third queue to the multiplier when the cell operates in the write mode.

20. The decoder of claim 1, wherein the summation logic includes an array of XOR gates.

21. The decoder of claim 1, wherein the inverse of the error location matrix is generated by firmware or hardware.

22. The decoder of claim 1, wherein the decoder is a component of a data storage device, and wherein the data blocks and one or more redundancy blocks are retrieved from a storage medium and held in a data buffer in the data storage device.

23. A data storage device configured to correct errors in date retrieved from a storage medium, comprising:

a data buffer configured to hold data blocks and one or more redundancy blocks retrieved from the storage medium; and

a decoder connected to the data buffer, the decoder having:

a plurality of cells that access the data blocks and one or more redundancy blocks held in the data buffer, each cell configured to generate a partial syndrome based on the retrieved data blocks and one or more redundancy blocks, and to generate a partial error value based on the partial syndrome and a portion of an inverse of an error location matrix, and

a summation logic connected to the plurality of cells, the summation logic configured to generate an error value based on the partial error values generated by the plurality of cells, wherein the error value corrects errors in a data block.

24. The data storage device of claim 23, wherein the number of cells in the plurality of cells is equal to the number of redundancy blocks retrieved from the storage medium.

25. The data storage device of claim 23, wherein each cell includes:

an input logic;

a first queue connected to the input logic;

a multiplier connected to the input logic and the first queue;

a second queue connected to the multiplier, wherein the second queue holds a root of a generator polynomial; and

a third queue connected to the multiplier, wherein the third queue holds the portion of the inverse of the error location matrix.

26. The data storage device of claim 25, wherein each cell operates in a read mode to generate the partial syndrome, and operates in a write mode to generate the partial error value.

27. The data storage device of claim 26, wherein when a cell operates in the read mode:

the input logic of the cell generates a sum of inputs received from the multiplier and the data buffer, and stores the sum in the first queue; and

the multiplier multiplies the root of a generator polynomial from the second queue and the sum from the first queue.

28. The data storage device of claim 27, wherein when the cell operates in the write mode:

the input logic of the cell receives the sum from the first queue; and

the multiplier multiplies the sum held in the queue and an element of the portion of the inverse of the error location matrix held in the third queue.

29. The data storage device of claim 28, wherein each cell in the plurality of cells is set to receive a different root of a generator polynomial to generate different partial syndromes, and a different portion of the inverse of the error location matrix to generate different partial error values.

30. The data storage device of claim 28, wherein the multiplier comprises a plurality of Galois Field multipliers to perform multiple, parallel Galois Field multiplications.

31. The data storage device of claim 23, wherein the summation logic includes an array of exclusive-or (XOR) gates.

32. The data storage device of claim 23, wherein the inverse of the error location matrix is generated by firmware or hardware.

33. A method of correcting errors in data, comprising:

receiving data blocks and one or more redundancy blocks;

generating an inverse of an error location matrix with locations of any received data blocks with errors;

generating a plurality of partial syndromes in a plurality of cells of a decoder, wherein each one of the plurality of partial syndromes are generated in each one of the plurality of cells based on the received data blocks and one or more redundancy blocks;

generating a plurality of partial error values in the plurality of cells, wherein each one of the plurality of partial error values are generated in each one of the plurality of

cells based on a partial syndrome and a portion of the inverse of the error location matrix; and

generating an error value based on the plurality of partial error values generated by the plurality of cells.

34. The method of claim 32, wherein the received data blocks and one or more redundancy blocks are held in a data buffer, and wherein generating a plurality of partial syndromes comprises:

in each of the plurality cells:

a) reading data from the data buffer;

b) generating a sum of the data read from the data buffer with an output from a multiplier;

c) storing the sum;

d) multiplying the sum with a root of a generator polynomial; and

e) iterating steps a) to d) to process all the data blocks and one or more redundancy blocks held in the data buffer, wherein the sum is the partial syndrome when all the data blocks and one or more redundancy blocks held in the data buffer have been processed.

35. The method of claim 34, wherein the number of iterations of steps a) to d) of claim 34 is equal to the number of data blocks and redundancy blocks held in the data buffer.

36. The method of claim 34, wherein in iterating step a), when data is to be read from a data block having errors, reading all zeros rather than the data from the data block having errors.

37. The method of claim 34, wherein multiplying the sum with a root of a generator polynomial comprises:

performing multiple, parallel Galois Field multiplications between the root of a generator polynomial and the sum using a plurality of Galois Field multipliers.

38. The method of claim 34, wherein generating a plurality of partial error values comprises:

after steps a) to e) of claim 34 are completed, in each of the plurality cells:

f) multiplying the partial syndrome with an element of the portion of the inverse of the error location matrix to generate a partial error value; and

g) sending the partial error value to be summed with the partial error values generated by each of the plurality of cells.

39. The method of claim 38, further comprising:

h) summing the partial error values generated by each of the plurality of cells to generate an error value; and

i) writing the error value to the data buffer to correct a data block having errors.

40. The method of claim 39, further comprising:

j) repeating steps f) and g) of claim 38 and steps h) and i) of claim 39 to correct another data block having errors.

41. The method of claim 40, wherein the number of iterations of steps f) to g) of claim 38 and steps h) and i) of claim 39 is equal to the number of data blocks having errors.

42. The method of claim 40, wherein the number of iterations of steps f) to g) of claim 38 and steps h) and i) of claim 39 is equal to the number of redundancy blocks held in the data buffer.

43. The method of claim 40, wherein the sum is cleared and steps a) to e) of claim 34, steps f) and g) of claim 38, steps h) and i) of claim 39, and step j) of claim 40 are repeated to process another portion of the data blocks and one or more redundancy blocks held in the data buffer.

44. The method of claim 38, wherein multiplying the partial syndrome with an element of the portion of the inverse of the error location matrix comprises:

performing multiple, parallel Galois Field multiplications between the partial syndrome and the element of the portion of the inverse of the error location matrix using a plurality of Galois Field multipliers.

45. The method of claim 33, wherein receiving data blocks and one or more redundancy blocks comprises retrieving data blocks and one or more redundancy blocks from a storage medium of a data storage device.

46. The method of claim 33, wherein the inverse of the error location matrix is generated by firmware or hardware.